Kinship and pedigree analysis: Methods and applications

Magnus Dehli Vigeland and Thore Egeland

Solutions for exercise set VI: Pedigree reconstruction

```
library(pedsuite)
library(pedbuildr)
```

Exercise VI-1

a) 100 markers, all SNPs with alleles 1 and 2. The genotype of the 1st person for the 3rd marker is 1/1.

b) R code given in the exercise.

c) The pairwise estimates below show that 2 and 3 are parent-child (in some order), and that 1 and 2 most likely are full siblings. The relationship between 1 and 3 is more distant.

```
kap = ibdEstimate(x)
showInTriangle(kap, labels = T, pos = c(1,3,2))
```
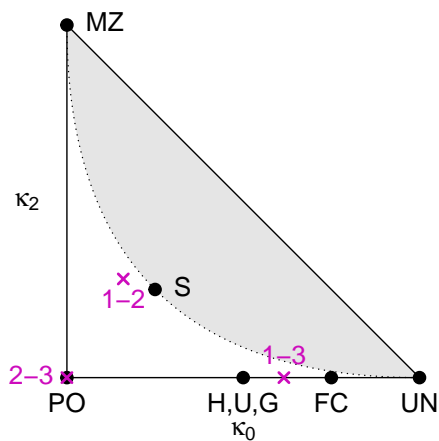

d) Assume that 1 and 2 are full siblings. Since 3 has a father/son-relationship with 2, but *not* with 1, the only possibility is that 3 is the son of 2. This implies that 3 is the nephew of 1, which is compatible with their point in the triangle above. Thus we are left with this pedigree:
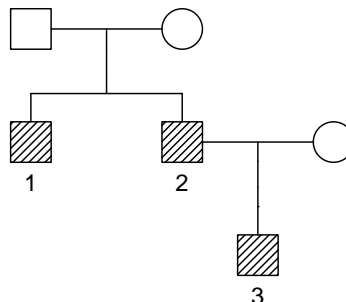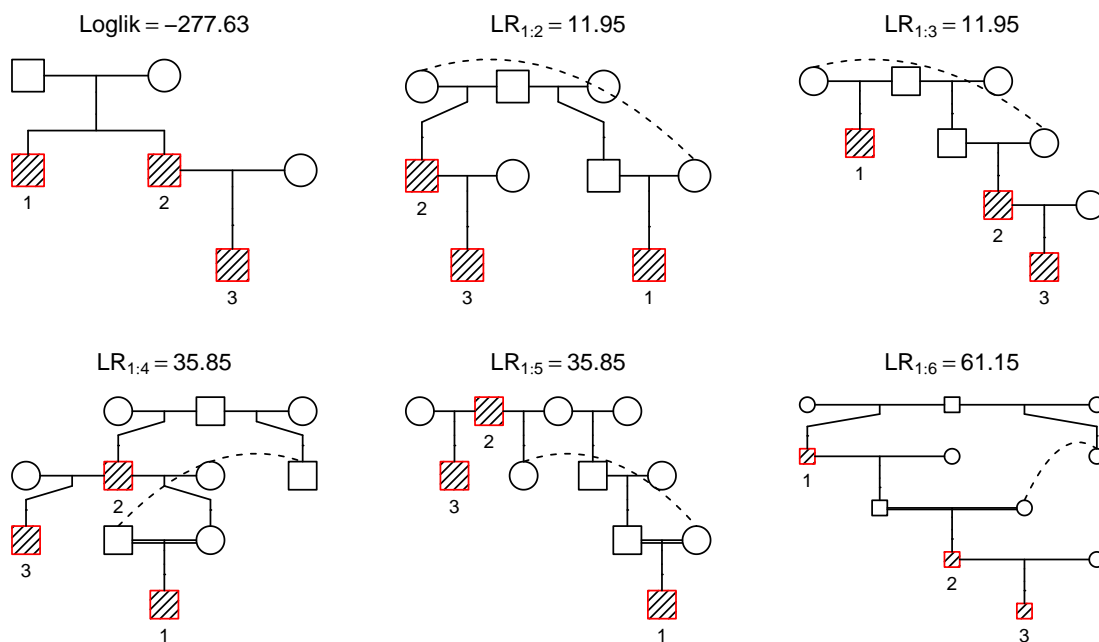
Exercise VI-2

a) The most likely pedigree is the same as we deduced by hand, ca 156 times more likely than the next.
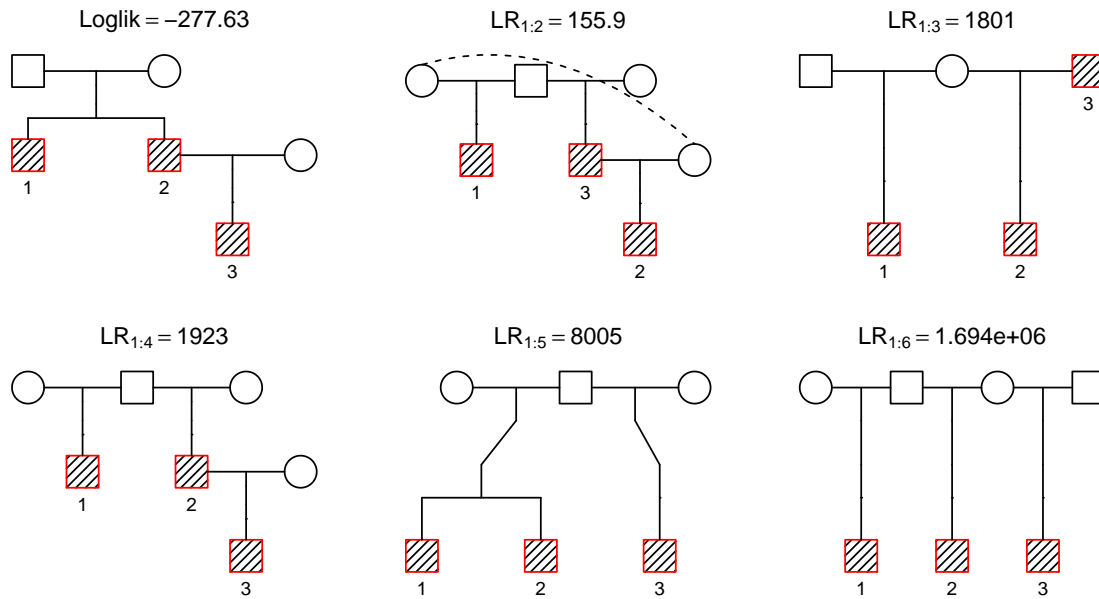
b) Note the combination of `knownPO` and `age` to indicate that 2 is the father of 3:

```r
r2 = reconstruct(x, extra = 3, knownPO = list(2:3), age = "2 > 3")
plot(r2, top = 6)
```


The winner is the same as before, but several new (rather esoteric) runners-up are included.

c) They are simultaneous half-avuncular and half first cousins:

```
verbalise(r2[[2]], ids = c(1,3))
```

```
## Half-avuncular: 1 is a half-uncle of 3
##     1-[e1]-2-3
## Half first cousins
##     1-e2-[e3]-2-3
```

Exercise VI-3

a) 24 markers, as reported by `summary()`:

```
x = readPed("data/reconstruct-fathers.ped")
x = setFreqDatabase(x, "data/reconstruct-fathers.freq")
```

```
summary(x)
```

```
## List of 5 singletons.
## Labels: 1 (female), 2 (female), 3 (male), 4 (male), 5 (male).
## 24 attached markers.
```

b) Here is one way to do it:

```
r = reconstruct(x, noChildren = 3:5, knownPO = list(c(1,3), c(2,4), c(2,5)))
```

c) All three children have the same father! Indeed, this is supported by both the two most likely pedigrees:

```
plot(r, top = 6)
```